

Optimasi Proses *Load* Aplikasi HDD Pada SO Windows Dengan *Prefetch Files*

Agustinus Suradi^{1*)}, Doni Setyawan²

^{1,2}Fakultas Ilmu Komputer, Universitas Widya Dharma Klaten

^{1,2}Jl. Ki Hajar Dewantara No.1 Klaten, PO.BOX.1168 Kode Pos.57438, Jawa Tengah, Indonesia

email: ¹simpati2000@mailcity.com, ²donisoft@gmail.com.

Received: 18 April 2018; Revised: 9 Mei 2018; Accepted: 13 Mei 2018

Copyright ©2018 Politeknik Harapan Bersama Tegal. All rights reserved

Abstract – In the current era with the help of IT devices can be done more quickly, effectively and efficiently. Good collaboration between brainware, software and hardware, will produce the expected work performance and response. The Windows operating system, when loading applications, analyzes the files used during startup and what applications are running, then create index files and save them in HDD. With the Windows index can load applications faster. Windows 7 will crack the activity, and the mapping will be reused while loading the HDD app so it will speed up system performance, try to load this application by using 3 prefetch files network method by disabling prefetch, clear prefetch and prefetch activation. Recording time required in HDD application load on Windows operating system with Prefetch Files, in a WMPLAYER.EXE application with 61,800 milliseconds, WORDPAD.EXE record time of 77,367 milliseconds, and CHROME.EXE with the record time of 904,400 milliseconds. Pada Using Prefetch Files then the load time of the HDD app will be faster than without using Prefetch Files or by removing Prefetch Files.

Abstrak – Pada era saat ini dengan bantuan perangkat IT pekerjaan dapat dilakukan lebih cepat, efektif dan efisien. Kolaborasi yang baik antara brainware, software dan hardware, akan menghasilkan performa dan respon kerja yang diharapkan. Sistem operasi Windows, ketika loading aplikasi, menganalisa file yang digunakan selama startup dan aplikasi-aplikasi apa yang dijalankan, lalu membuat index file dan menyimpannya dalam HDD. Dengan index tersebut Windows dapat load aplikasi lebih cepat. Prefetch Files pada Windows akan mentracking aktivitas yang terjadi, dan pemetaan tersebut akan dipergunakan kembali sewaktu load aplikasi HDD sehingga akan mempercepat kinerja sistem, pengujian load aplikasi ini dilakukan dengan 3 metode pengelolaan prefetch files yaitu dengan disable prefetch, clear prefetch dan aktif prefetch. Hasil pencatatan waktu yang diperlukan dalam load aplikasi HDD pada sistem operasi windows 7 ultimate dengan Prefetch Files, pada aplikasi WMPLAYER.EXE dengan catatan waktu 61,800 milliseconds, WORDPAD.EXE dengan catatan waktu 77,367 milliseconds, dan CHROME.EXE dengan catatan waktu 904,400 milliseconds. Pada Penggunaan Prefetch Files maka waktu load aplikasi HDD akan lebih cepat dibandingkan dengan tanpa menggunakan Prefetch Files atau dengan menghapuskan Prefetch Files.

*) **Corresponding author:** Agustinus Suradi
Email: simpati2000@mailcity.com

Kata Kunci – Load applications, HDD, Prefetch Files, Operating System, Windows.

I. PENDAHULUAN

Penggunaan personal komputer dan perangkat *IT* sebagai pendukung dalam menyediakan dan mengolah data menjadi informasi yang cepat dan efisien merupakan hal yang tidak terpisahkan pada era *IT*. Interaksi antara manusia dan komputer memiliki tiga komponen penting yaitu manusia komputer dan interaksi, ketiga komponen tersebut saling mendukung dan berkaitan, dalam mencapai sinerginya [1].

Komputer yang digunakan dalam suatu periode tertentu dan tidak dilakukan optimasi kinerjanya semakin lambat, penambahan jumlah aplikasi yang diinstal, pengoperasian aplikasi yang bersamaan dalam sistem operasi juga akan berdampak pada kinerja *PC*. Perangkat keras seperti *HDD* memiliki peranan penting dalam komputer karena tanpa ruang kosong yang cukup, akan berpengaruh terhadap waktu akses data dan *loading* aplikasi. Sehingga sistem komputer dapat dipastikan semakin lama digunakan pasti kinerjanya semakin menurun [2][3][4]

Kemampuan untuk menyediakan dan mengakses informasi secara cepat dan akurat menjadi sangat esensial bagi sebuah organisasi [5]. Ketika sistem komputer menurun *performancenya*, ada beberapa cara yang bisa dilakukan untuk optimasi sistem operasi *Windows*, baik dengan peningkatan komponen *hardware*, maupun dengan pengelolaan *software* [3][4]. Apabila dari sisi *hardware* tidak memungkinkan maka peningkatan pengelolaan *software* yang perlu kita perhatikan. Pada beberapa forum di media sosial sering dibahas tentang peningkatan kinerja dari sebuah sistem operasi *Windows7*. Salah satunya yakni cara untuk mempercepat loading aplikasi pada sistem operasi *Windows* dengan pengelolaan *prefetch files* [6][7][8][9].

Dengan latar belakang tersebut maka penulis akan mengkaji pengaruh dari kinerja *Windows* yang berkaitan dengan pengelolaan *prefetch files*. Penulis ingin memberikan informasi yang pasti mengenai waktu yang dibutuhkan untuk load aplikasi yang berkaitan dengan *prefetch files*. Sehingga nantinya diharapkan dapat dijadikan panduan untuk pengelolaan *prefetch files*, dan dengan serangkaian pengujian untuk membuktikannya.

II. PENELITIAN YANG TERKAIT

Sistem Operasi merupakan perangkat lunak yang bertugas melakukan kontrol, manajemen perangkat keras, dan juga operasi-operasi dasar sistem, menentukan waktu dan prioritas program yang akan dijalankan, dan *scheduling* perangkat (*memory* atau *I/O*) yang akan yang digunakan, sehingga aplikasi-aplikasi yang dijalankan dapat berproses dengan lancar, dan memudahkan kegiatan-kegiatan yang dilakukan oleh *users* [3].

Penelitian yang dilakukan oleh Timothy Highley dan Paul Reynolds, pada 2003 dengan judul “*Marginal Cost-Benefit Analysis for Predictive File Prefetching*” penelitian ini menguraikan bahwa *prefetching* sangat berperan dalam menentukan akses *software on-line*, *Prefetching* melibatkan dan memprediksi akses yang akan dikerjakan dan menetapkan kapan / apakah akan menggunakan *file prefetch*. Analisis biaya-manfaat (*Cost-Benefit Analysis / CBA*) membahas kapan / apakah akan melakukan *prefetch*, serta membahas interaksi antara *prefetching* dan *caching* [10]. Pentingnya penggunaan teknologi dengan jumlah perangkat yang digunakan dapat diminimalisir namun dengan kinerja yang lebih optimal [11].

Penelitian yang dilakukan oleh Pei Cao dkk. pada 1996 dengan judul “*Caching, Prefetching, And Disk Scheduling*” artikel ini menyajikan desain, implementasi, dan kinerja sistem *file* yang mengintegrasikan *caching* aplikasi-dikendalikan, *prefetching*, dan penjadwalan disk. Kami menggunakan strategi manajemen *cache dua level*. Kernel menggunakan kebijakan *LRU-SP (Least-Recently-Used with Swapping dan Placeholder)* untuk mengalokasikan pemblokiran ke proses, dan setiap proses mengintegrasikan *caching* dan pemuatan khusus aplikasi berdasarkan kebijakan yang dikontrol-agresif, suatu algoritma yang sebelumnya ditunjukkan dalam teoritis akal untuk menjadi hampir optimal [12].

Setiap proses juga meningkatkan latensi akses *disknya* dengan mengirimkan *prefetch*-nya dalam *batch* sehingga permintaan dapat dijadwalkan untuk mengoptimalkan kinerja akses *disk*. Pengukuran ini menunjukkan bahwa kombinasi teknik ini sangat meningkatkan kinerja sistem *file* [13][14][15]. Pada penelitian ini membahas tentang optimasi proses *load* aplikasi *HDD* dengan pengelolaan *Prefetch Files*, sehingga didapat hasil yang dapat meningkatkan kecepatan optimal proses *load* aplikasi *HDD*.

III. METODE PENELITIAN

A. Metode

Metode dasar yang digunakan dalam penelitian ini adalah penelitian eksperimen. Metode analisis data dengan menggunakan kerangka kerja yang dilakukan dengan beberapa tahap yaitu:

1) *Tahap 1 - Pemahaman Awal (Gain Understanding)*: Tahap ini akan dilakukan deskripsi dan identifikasi terhadap hipotesa untuk menentukan proses-proses apa saja yang akan dilakukan, dari rumusan masalah, dan tujuan dari penelitian yang dilakukan.

2) *Tahap 2 – Study Literature*: Dalam tahap ini dilakukan kajian pustaka, untuk mendapatkan landasan teori mengenai masalah yang akan diteliti, baik dari buku-buku dan sumber referensi lainnya.

3) *Tahap 3 – Desain Penelitian*: Pada tahap ini dilakukan untuk proses penentuan desain penelitian, penentuan variabel, *tools*, proses pengolahan data, dan analisisnya.

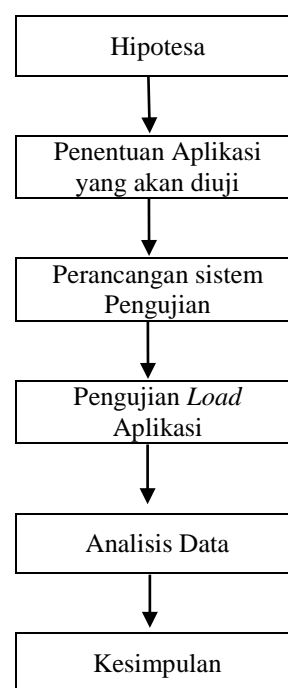
4) *Tahap 4 – Pengujian load aplikasi*: Tahap ini dilakukan proses pengujian *load* aplikasi dan pengelolaan *prefetch files*.

5) *Tahap 5 – Tahap Analisis Data*: Dalam tahap ini dilakukan analisa terhadap *load* aplikasi yang telah dilakukan pengujian untuk selanjutnya dibuat kesimpulan dan rekomendasi. Rekomendasi ini dapat membantu untuk pengelolaan *prefetch files*.

6) *Tahap 6 – Dokumentasi*: Penyusunan laporan untuk mendokumentasikan hasil penelitian dan kesimpulan.

B. Alur Penelitian

Berdasarkan tahapan penelitian yang akan dilakukan maka dibuatlah alur penelitian dari proses optimasi *load* aplikasi:



Gbr. 1 Alur Penelitian

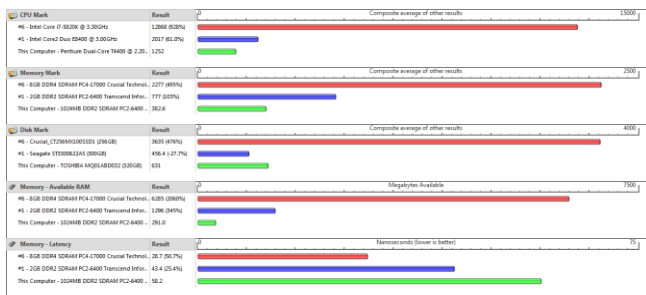
IV. HASIL DAN PEMBAHASAN

A. Test Performance

Perangkat yang digunakan untuk *load* aplikasi terlebih dahulu diidentifikasi, meliputi *system hardware*, CPU dan *memori information* dengan *Performance Test 8.0*. Berikut di bawah ini teridentifikasi hasil test menggunakan *Performance Test 8.0* yang disertakan dua *reference performance* sebagai berikut pada Gbr.2.

Item	This Computer	#6	#1
Performance Test Information			
Performance Test Version	8.0 (1045) WM832	8.0 (1045) WM864	8.0 (102) WM832
PassMark Rating	326	4995	1060
System Information			
System Name	ORTODETO1	REFERENCE-6	REFERENCE-1
Model			
Operating System	Windows 7 Ultimate Edition build ...	Windows 8.1 Professional Edition ...	Windows XP Professional Service P...
Motherboard Manufacturer	Heavett-Packard	Gigabyte Technology Co., Ltd.	Intel Corporation
Motherboard Model	3607	X99-UD4-CF	DP55DP
Motherboard Version	99.BF	x.x	AAD81073_207
BIOS Manufacturer	Insyde	American Megatrends Inc.	Intel Corp.
BIOS Version	F.60	ALASKA - 1072009	INTEL - 23c
BIOS Release Date	2009/10/22	2014/09/18	2009/07/15
CPU Information			
Manufacturer	GenuineIntel	GenuineIntel	GenuineIntel
Type	Pentium Dual-Core T4400 @ 2.20GHz	Intel Core i7-5820K @ 3.30GHz	Intel Core2 Duo E8400 @ 3.00GHz
Codename	Penryn	Haswell-E	Wolfdale
CPUID	Family 6, Model 17, Stepping A, Re...	Family 6, Model 3F, Stepping 2, Re...	Family 6, Model 17, Stepping 6, Re...
Socket	Socket P (478)	LGA2011-v3	LGA775
Lithography	45nm	22nm	45nm
Number of CPU's	1	1	1
Cores per CPU	2	6	2
Logicals per Core	1	1	1
Clock Frequencies			
Measured Speed	2194.9 MHz	3200.5 MHz (Turbo: 3600.6 MHz)	3000.0 MHz
Multiplier	11.0X	33.0X	9.0X
Bus Speed	199.5 MHz	100.0 MHz	333.3 MHz
Front Side Bus Speed	7908MHz	(N/A)	1333MHz
Timing Error Ratio	0.990	1.002	1.000
Cache per CPU package			
L1 Instruction Cache	2 x 32 KB	6 x 32 KB	2 x 32 KB
L1 Data Cache	2 x 32 KB	6 x 32 KB	2 x 32 KB
L2 Cache Size	1 x 1024 KB	6 x 256 KB	1 x 6 MB
L3 Cache	(N/A)	15 MB	(N/A)
Memory Information			
Total Physical Memory	955MB	8018MB	2029MB
Available Physical Memory	238MB	6319MB	1278MB
Memory Devices			
Slot 1	1GB DDR2 SDRAM PC2-6400	4GB DDR4 SDRAM PC4-17000	1GB DDR2 SDRAM PC2-6400
	Hynix Semiconductor (Hyundai Ele...	Crucial Technology CT4G4DFS213...	Transcend Information JM800QLU...
	SS1L 1.8V, Clk: 400.0MHz, Timings 6...	1.2V, Clk: 1066.7MHz, Timings 15...	SS1L 1.8V, Clk: 400.0MHz, Timings 5...

Gbr. 2 Hasil Performance Test



Gbr. 3 Hasil benchmark CPU, memory and HDD dengan PerformanceTest

ReadyBoot	1/24/2017 9:21	File folder
AgAppLaunch	8/28/2015 1:27	Data Base File
AgCx_SC1	11/27/2016 8:35	Data Base File
AgCx_SC1.db.trx	11/27/2016 8:34	TRX File
AgCx_SC2	1/24/2017 2:18	Data Base File
AgCx_SC3_5FB0CEE7	1/4/2017 10:58	Data Base File
AgGIFaultHistory	8/19/2017 9:36	Data Base File
AgGIFAppHistory	8/19/2017 9:36	Data Base File
AgGIGlobalHistory	8/19/2017 9:36	Data Base File
AgGIUAD_P_S-1-5-21-3047235092-21...	1/24/2017 2:27	Data Base File
AgGIUAD_S-1-5-21-3047235092-2173...	1/24/2017 2:27	Data Base File
AgRobust	8/19/2017 9:36	Data Base File
AUDIODG.EXE-BDFD3029.pf	1/24/2017 9:18	PF File
AUTORUN.EXE-2F753DD6.pf	1/22/2017 6:29	PF File
CHROME.EXE-5A1054AF.pf	1/24/2017 2:39	PF File
CMD.EXE-4A818364.pf	1/24/2017 5:25	PF File
CONHOST.EXE-1F3E9D7E.pf	1/24/2017 7:39	PF File
CONSENT.EXE-531B09EA.pf	1/24/2017 9:18	PF File
CORELDRW.EXE-3152B5FC.pf	1/24/2017 8:05	PF File

Gbr. 4 View folder prefetch files

Hasil benchmark dengan PerformanceTest meliputi CPU Mark, Memory Mark, Disk Mark, Memory - Available RAM dan Memory - Latency menunjukkan data seperti pada Gbr. 3.

B. Konten Prefetch Files

File prefetch pada sistem operasi windows 7 ultimate 32 bit terdapat pada direktori C:\Windows\Prefetch, dapat kita buka dengan menggunakan windows explorer, menuju ke folder prefetch tersebut, ketika proses dilakukan maka tampil jendela persetujuan. Hasil dari proses continue diatas adalah didapat view dengan kondisi seperti di bawah ini, isi dari folder prefetch files, yang di dalamnya terdapat folder ReadyBoot dan komponen file yang lain.

Dalam prefetch files terdapat: jejak booting, aplikasi dan aplikasi hosting dan masing masing jenis prefetch files memiliki tujuan independen. Tujuan utama jejak boot prefetch files adalah untuk membantu mempercepat sistem operasi saat itu sedang dimulai atau pada waktu reboot. File aplikasi prefetch diciptakan dengan tujuan untuk mempercepat waktu yang dibutuhkan oleh Windows untuk memuat aplikasi tertentu tersebut. Aplikasi ini mencakup aplikasi bawaan Windows misalnya WMPLAYER.EXE, WORDPAD.EXE maupun aplikasi lain yang berjalan di windows. Misalnya CHROME.EXE. File aplikasi prefetch hosting, yang mencatat aktivitas jejak program tertentu yang digunakan untuk proses sistem, program ini misalnya Dllhost.exe, rundll.exe dan MMC.exe. Windows membutuhkan cara untuk melacak program yang berbeda dengan proses yang berbeda, yang dikategorikan secara terpisah sebagai aplikasi hosting.

Prefetch files dapat dilihat hasil run dari software utility dengan menggunakan WinPrefetchView v1.35, produk Nir Sofer dengan view diperlihatkan pada Gbr.5.

Filename	Created Time	Modified Time	File Size	Process EXE	Process Path	Run Counter	Last Run Time	Missing Pr...
WINRAR.EXE-94E7D80C.pf	1/19/2017 2:21:49	1/24/2017 12:45:54	83,266	WINRAR.EXE	C:\PROGRAM FILES\MICROSOFT OFFICE\...	7	1/24/2017 12:45:50	No
WINWORD.EXE-C91725A1.pf	1/16/2017 7:10:34	1/22/2017 9:11:57	195,958	WINWORD.EXE	C:\PROGRAM FILES\MICROSOFT OFFICE\...	8	1/22/2017 9:11:45	No
WMADAP.EXE-F8DFDFA2.pf	8/28/2015 1:22:41	1/24/2017 2:39:05	79,988	WMADAP.EXE	C:\Windows\System32\wbem\WMADAP..._1.133	1,133	1/24/2017 2:39:48	No
WMMPRVSE.EXE-1628051C.pf	8/28/2015 1:19:18	1/24/2017 2:39:00	40,188	WMMPRVSE	C:\Windows\System32\wbem\WMADAP..._1.133	2,357	1/24/2017 2:38:49	No
WMPLAYER.EXE-BAD6BD53.pf	1/19/2017 8:19:04	1/19/2017 8:55:38	130,302	WMPLAYER.EXE	C:\Windows\System32\WMPLA...	2	1/19/2017 8:55:37	No
WMPNETWK.EXE-D9F2A96F.pf	1/27/2016 7:47:53	1/24/2017 7:32:23	56,264	WMPNETWK.EXE	C:\PROGRAM FILES\WINDOWS MEDIA PL...	443	1/24/2017 12:32:42	No
WMPNSCFG.EXE-FCD039BF.pf	1/17/2017 12:44:12	1/19/2017 4:09:53	24,122	WMPNSCFG.EXE	C:\PROGRAM FILES\WINDOWS MEDIA PL...	32	1/19/2017 4:09:53	No
WUUAUCLT.EXE-70318591.pf	1/11/2017 9:37:28	1/24/2017 7:40:23	36,488	WUUAUCLT.EXE	C:\Windows\System32\WUUAUCLT...	20	1/24/2017 7:40:13	No
WUDFHOST.EXE-AFFEF87C.pf	7/16/2016 2:34:50	1/23/2017 7:31:22	22,388	WUDFHOST.EXE	C:\Windows\System32\WUDFHOST...	499	1/23/2017 7:31:11	No

Gbr. 5 View prefetch files dengan WinPrefetch

Filename	Created Time	Modified Time	File Size	Process EXE	Process Path	Run Counter	Last Run Time	Missing Pro...
WINPREFETCHVIEW.EXE-B8112189.pf	1/24/2017 2:39:05	1/24/2017 2:39:05	83,210	WINPREFETCHVIEW.EXE	C:\Windows\System32\prefetch...	1	1/24/2017 2:39:05	No
WINRAR.EXE-94E7D80C.pf	1/19/2017 9:21:49	1/24/2017 7:45:14	83,266	WINRAR.EXE	C:\PROGRAM FILES\MICROSOFT OFFICE\...	7	1/24/2017 7:45:14	No
WINWORD.EXE-C91725A1.pf	1/16/2017 7:10:34	1/22/2017 9:11:57	195,958	WINWORD.EXE	C:\PROGRAM FILES\MICROSOFT OFFICE\...	8	1/22/2017 9:11:57	No
WMADAP.EXE-F8DFDFA2.pf	8/28/2015 1:22:41	1/24/2017 7:33:49	79,988	WMADAP.EXE	C:\Windows\System32\wbem\WMADAP..._1.133	1,133	1/24/2017 7:33:49	No
WMMPRVSE.EXE-1628051C.pf	8/28/2015 1:19:18	1/24/2017 9:19:00	40,188	WMMPRVSE	C:\Windows\System32\wbem\WMADAP..._1.133	2,357	1/24/2017 9:19:00	No
WMPLAYER.EXE-BAD6BD53.pf	1/19/2017 8:19:04	1/19/2017 8:55:38	130,302	WMPLAYER.EXE	C:\Windows\System32\WMPLA...	2	1/19/2017 8:55:38	No
WMPNETWK.EXE-D9F2A96F.pf	1/27/2016 7:47:53	1/24/2017 7:32:23	56,264	WMPNETWK.EXE	C:\PROGRAM FILES\WINDOWS MEDIA PL...	443	1/24/2017 7:32:23	No
WMPNSCFG.EXE-FCD039BF.pf	1/17/2017 12:44:12	1/19/2017 4:09:53	24,122	WMPNSCFG.EXE	C:\PROGRAM FILES\WINDOWS MEDIA PL...	32	1/19/2017 4:09:53	No
WUUAUCLT.EXE-70318591.pf	1/11/2017 9:37:28	1/24/2017 7:40:23	36,488	WUUAUCLT.EXE	C:\Windows\System32\WUUAUCLT...	20	1/24/2017 7:40:23	No
WUDFHOST.EXE-AFFEF87C.pf	7/16/2016 2:34:50	1/24/2017 2:31:22	22,388	WUDFHOST.EXE	C:\Windows\System32\WUDFHOST...	499	1/24/2017 2:31:22	No

Gbr. 6 View prefetch files WMPLAYER.EXE-BAD6BD53.pf dengan WinPrefetch.

Pada saat komputer dihidupkan, SO Windows 7 akan melacak cara komputer mulai dan program apa yang akan dan sering dibuka, Windows akan menyimpan informasi ini dalam folder prefetch, Windows akan mentracking aktivitas yang terjadi, Terdapat 128 entri dalam folder prefetch.

C. Load Aplikasi

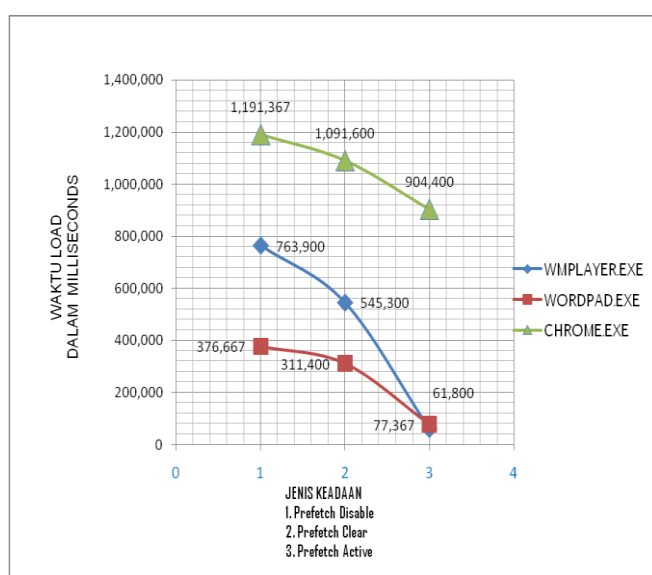
File aplikasi prefetch menghasilkan nilai, konvensi penamaan file prefetch ini menggunakan nama aplikasi yang dieksekusi dan ekstensi, diikuti oleh bit hash tiga puluh dua atau nomor nomor yang diwakili dalam heksadesimal, dengan ".pf" ekstensi. Pada waktu windows load aplikasi misalnya WMPlayer.exe prefetch akan menampung aktivitas tersebut. Hasil prefetch files ketika WMPlayer.exe diload pada prefetch

files yakni WMPLAYER.EXE-BAD6BD53.pf, seperti terlihat pada Gbr.6.

Pada pengujian load aplikasi yang dilakukan terhadap WMPLAYER.EXE, WORDPAD.EXE dan GOOGLE CHROME.EXE didapat catatan beberapa waktu yang diperoleh, diambil nilai rata-rata setiap load aplikasi HDD seperti ditunjukkan pada TABEL I.

TABEL I
HASIL AVERAGE LOAD DALAM MILLISECONDS

Aplikasi	Average Waktu Dalam MILLISECONDS		
	Prefetch Disable	Prefetch Clear	Prefetch Active
WMPLAYER.EXE	763,900	545,300	61,800
WORDPAD.EXE	376,667	311,400	77,367
CHROME.EXE	1.191,367	1.091,600	904,400



Gbr. 7 Waktu load aplikasi HDD

Hasil pengujian load aplikasi HDD diatas dapat dinyatakan bahwa pada aplikasi:

- WMPLAYER.EXE dengan kondisi Prefetch Disable kecepatan load aplikasi HDD 763,900 ms, kondisi Prefetch Clear kecepatan load aplikasi HDD 545,300 ms, dan kondisi Prefetch Active kecepatan load aplikasi HDD 61,800 ms.
- WORDPAD.EXE dengan kondisi Prefetch Disable kecepatan load aplikasi HDD 376,667 ms, kondisi Prefetch Clear kecepatan load aplikasi HDD 311,400 ms, dan kondisi Prefetch Active kecepatan load aplikasi HDD 77,367 ms.
- CHROME.EXE dengan kondisi Prefetch Disable kecepatan load aplikasi HDD 1.191,367 ms, kondisi Prefetch Clear kecepatan load aplikasi HDD 1.091,600 ms, dan kondisi Prefetch Active kecepatan load aplikasi HDD 904,400 ms.

Pada Penggunaan Prefetch Files maka waktu load aplikasi HDD akan lebih cepat dibandingkan dengan tanpa

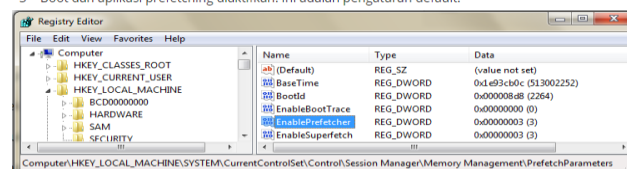
menggunakan Prefetch Files atau dengan menghapuskan Prefetch Files. Dari hasil diatas dapat divisualisasikan dalam garfik pada Gbr.7.

D. Optimasi Prefetch Files

Pengelolaan prefetch files dapat dilakukan dengan beberapa cara pada kondisi default prefetch files dalam kondisi aktif. Untuk mendukung kecepatan load aplikasi dalam Windows 7 disarankan agar prefetch files aktif. Dalam kondisi tertentu Prefetch files dapat dikontrol dengan cara manual maupun dengan bantuan software utility prefetch files. Pada penggunaan software utility Tweak Prefetch untuk mendukung kecepatan load aplikasi silahkan pada opsi panel Prefetch dipilih untuk mengaktifkan Applications atau dapat dipilih Applications & Boot, juga pada panel Superfetch silahkan dipilih Applications & Boot.

Secara manual Prefetch Files dapat dikelola lewat Registry Editor, ada beberapa nilai kondisi yang bisa dimungkinkan terhadap fungsi Prefetch Files, kondisi tersebut adalah :

Ada 4 nilai yang valid untuk kunci EnablePrefetch:
 0 = Prefetch dinonaktifkan. Windows akan menonaktifkan prefetch jika SSD (Solid State Drive) terdeteksi.
 1 = Aplikasi prefetch. Menciptakan prefetch hanya pada peluncuran aplikasi.
 2 = Boot prefetching diaktifkan. Menciptakan prefetch untuk proses boot.
 3 = Boot dan aplikasi prefetching diaktifkan. Ini adalah pengaturan default.



Gbr. 8 Windows Registry Enable Prefetchhher

Nilai kondisi yang tercepat ketika load aplikasi adalah pada value = 3.

Pada windows registry, Enable Prefetchhher pada Value data di set pada nilai = 3. Pada Enable Prefetchhher ketika nilai diset pada kondisi 3 yang menjadi nilai default maka Windows akan mentracking aktivitas yang terjadi, dan pemetaan tersebut akan dipergunakan kembali sewaktu load aplikasi HDD sehingga akan mempercepat kinerja sistem.

V. KESIMPULAN

Kesimpulan yang dapat diperoleh dari hasil penelitian ini adalah waktu load aplikasi HDD WMPLAYER.EXE dengan kondisi Prefetch Disable kecepatan load aplikasi HDD 763,900 ms, kondisi Prefetch Clear kecepatan load aplikasi HDD 545,300 ms, dan kondisi Prefetch Active kecepatan load aplikasi HDD 61,800 ms. WORDPAD.EXE dengan kondisi Prefetch Disable kecepatan load aplikasi HDD 376,667 ms, kondisi Prefetch Clear kecepatan load aplikasi HDD 311,400 ms, dan kondisi Prefetch Active kecepatan load aplikasi HDD 77,367 ms. CHROME.EXE dengan kondisi Prefetch Disable kecepatan load aplikasi HDD 1.191,367 ms, kondisi Prefetch Clear kecepatan load aplikasi HDD 1.091,600 ms, dan kondisi Prefetch Active kecepatan load aplikasi HDD 904,400 ms.

Optimasi load aplikasi pada HDD tercapai jika nilai default EnablePrefetch pada Windows 7 diset pada kondisi nilai 3, yakni applications & boot prefetching aktif. Kondisi nilai yang lain yaitu nilai: 0, 1, dan 2. Jika nilai diset 0 maka

prefetch dinonaktifkan, atau *Windows* akan menonaktifkan *prefetch* jika *SSD (Solid State Drive)* terdeteksi. Jika nilai 1 maka *Applications prefetch* aktif, dan menciptakan *prefetch* pada peluncuran aplikasi. Jika nilai diset 2 maka *boot prefetching* diaktifkan, sehingga menciptakan *prefetch* untuk proses boot.

DAFTAR PUSTAKA

- [1] S. Agustinus, *Interaksi Manusia dan Komputer, I ed.* Yogyakarta, Indonesia: AG. Litera Yogyakarta, ISBN 9786023960743., 2016.
- [2] C. David, K. Ken, and P. Allan, "Software Prefetching," in *Proceedings of the Fourth International Conference on Architectural Support for Programming Languages and Operating Systems.* ASPLOS IV., New York, USA, 1991, pp. ACM: 40–52. doi: 10.1145/106972.106979. ISBN 0897913809.
- [3] S. William, *Operating Systems, Internals and Design Principles, Seventh Edition ed.*: Prentice Hall, Pearson Education, Inc., 2012.
- [4] S. Yan, *Fundamentals of parallel multicore architecture.*: FL: CRC Press, Taylor & Francis Group ISBN 1482211181, 2016.
- [5] S. Agustinus, J. P. Hendro, "Contingency Planning pada Website Universitas Widya Dharma," *Jurnal Teknologi Informasi Respati*, vol. Vol. X Nomor 29, ISSN: 1907-2430, Juli 2015.
- [6] S. Nir, *Win Prefetch View*, (2016, Desember) [Online]. https://www.nirsoft.net/utils/win_prefetch_view.html.
- [7] Infosec Institute. *Windows Systems and Artifacts in Digital Forensics: Part III: Prefetch Files*, (2017, Januari) [Online]. <http://resources.infosecinstitute.com/windows-systems-artifacts-digital-forensics-part-iii-prefetch-files>.
- [8] C.Tien-Fu And B. Jean-Loup, "Effective hardware-based data prefetching for high-performance processors," *IEEE Transactions on Computers*, vol. 44 , no. (5), pp. 609–623. doi: 10.1109/12.381947. ISSN 0018-9340., 1995.
- [9] M. Sparsh,, "A Survey of Recent Prefetching Techniques for Processor Caches," *ACM Comput. Surv*, vol. 49, pp. 35:1–35:35. doi:10.1145/2907071. ISSN 0360-0300., Feb. 2016.
- [10] R. Paul And H. Timothy, "Marginal Cost-Benefit Analysis for Predictive File Prefetching," *University of Virginia Charlottesville, , VA, USA.*, Technical Report 2003.
- [11] P. W. Ari, "Optimasi Jaringan Local Area Network Menggunakan VLAN dan VOIP," *Jurnal Informatika:Jurnal Pengembangan IT (JPIT)*, vol. Vol. 2, No. 1, Januari 2017.
- [12] C. Pei, F. Edward W, K. Anna R, nd L. Kai, "Implementation and performance of integrated application-controlled file caching, prefetching, and disk scheduling," *Journal ACM Transactions on Computer Systems (TOCS)*, vol. Volume 14 , no. Issue 4, pp. pp. 311-343, Nov 1996.
- [13] B. Jean-Loup And C. Tien-Fu, "An Effective On-chip Preloading Scheme to Reduce Data Access Penalty," in *ACM/IEEE Conference on Supercomputing*, ISBN 0897914597., New York, NY, USA, 1991-01-01, pp. 176–186 doi:10.1145/125826.125932.
- [14] Porterfield, And K Allan, "Software methods for improvement of cache performance on supercomputer applications," *Rice University, Houston, Texas, U.S, Thesis* 1989-01-01.
- [15] S. Palacharla And R. E Kessler, "Evaluating Stream Buffers As a Secondary Cache Replacement," in *Proceedings of the 21st Annual International Symposium on Computer Architecture. ISCA '94*. Los Alamitos, CA, USA: IEEE Computer Society Press, Los Alamitos, CA, USA, 1994-01-01, pp. 24–33. doi: 10.1145/191995. 192014. ISBN 0818655100.